A Fast Correlation Attack on the Shrinking Generator*

Bin Zhang[1,2], Hongjun Wu[1], Dengguo Feng[2], and Feng Bao[1]

[1] Institute for Infocomm Research, Singapore
[2] State Key Laboratory of Information Security,
Graduate School of the Chinese Academy of Sciences,
Beijing 100039, P.R. China
zhangbin@mails.gscas.ac.cn
{hongjun,baofeng}@i2r.a-star.edu.sg

Abstract. In this paper we demonstrate a fast correlation attack on the shrinking generator with known connections. Our attack is applicable to arbitrary weight feedback polynomial of the generating LFSR and comparisons with other known attacks show that our attack offers good trade-offs between required keystream length, success probability and complexity. Our result confirms Golić's conjecture that the shrinking generator may be vulnerable to fast correlation attacks without exhaustively searching through all possible initial states of some LFSR is correct.

Keywords: Fast correlation attack, Shrinking generator, Linear feedback shift register.

1 Introduction

The shrinking generator (SG) is a well-known keystream generator proposed in [4] at Crypto'93. It consists of two LFSR's, say LFSR A and LFSR S. Both LFSRs are regularly clocked and the output bit of the generating LFSR A is taken iff the current output bit of the control LFSR S is 1. This generator obtains a kind of implicit non-linearity from the shrinking process, i.e. the exact positions of the remaining bits in the generated keystream become uncertain. It is proved that the generated keystream has many merits in cryptographic sense such as a long period, a desirably high linear complexity and good statistical properties. It is recommended in [4] that both the initial states of the two LFSR's and the feedback polynomials of theirs be secret key. As in [5], we stress here that our analysis is also based on the known feedback polynomials assumption.

So far, several attacks against the shrinking generator have been proposed. A simple divide-and-conquer attack is proposed in [4] requiring an exhaustive search through all possible initial states and feedback polynomials of LFSR S. A

* Supported by National Natural Science Foundation of China (Grant No. 60273027), National Key Foundation Research 973 project (Grant No. G1999035802) and National Science Fund for Distinguished Young Scholars (Grant No. 60025205).

A.J. Menezes (Ed.): CT-RSA 2005, LNCS 3376, pp. 72–86, 2005.

correlation attack is proposed in [8] and is experimentally analyzed in [19] which takes an exhaustive search through all initial states and all possible feedback polynomials of LFSR A. At Asiacrypt'98, T. Johansson [12] presented a reduced complexity correlation attack based on searching for specific subsequences of the keystream, whose complexity and required keystream length are both exponential in the length of LFSR A. In 2001, a probabilistic correlation analysis [6] based on a recursive computation of the posterior probabilities of individual bits of LFSR A was conducted by J. D. Golić, which revealed the possibility of implementing certain fast correlation attack on the shrinking generator. A novel distinguishing attack on the shrinking generator is proposed in [5]. According to the facts that an arbitrary weight feedback polynomial of degree L is known to have a weight 4 multiple of degree $O(2^{L/3})$ and $10000 = 2^{13.2877} = 2^{L/3}$ [7, 20] , that distinguisher is applicable to arbitrary shrunken LFSR's of length around 40. Very recently, an improved linear consistency attack is presented in [17] which is an completely exhaustive search through all initial states of LFSR S.

In [6], it was conjectured that the shrinking generator *may* be vulnerable to fast correlation attacks that would not require an exhaustive search through all possible initial states of LFSRs. In this paper we try to answer this question definitely even for LFSR A of length 61, as suggested in [9]. We show that given a length of only 140000 keystream bits, the initial state of LFSR A with arbitrary weight feedback polynomial of degree 61 can be recovered with success probability higher than 99% and complexity 2^{56}, which is a good trade-off between these parameters.

This paper is organized as follows. In Section 2 we present a general description of our attack. Deep analysis of our attack is made in Section 3. Experiments results together with comparisons with other attacks on the shrinking generator are provided in Section 4. Finally, conclusions are given in Section 5.

2 A General Description of Our Attack

We first present a general description of our attack. Denote the output sequence of LFSR A by $a = a_0, a_1, \cdots$ and the output sequence of LFSR S by $s = s_0, s_1, \cdots$. The output keystream of (SG) is $z = z_0, z_1, \cdots$. Our attack is composed of two phases: first, correlation analysis phase which results in a sequence $\hat{a} = \hat{a}_0, \hat{a}_1, \cdots$ associated with sequence a by the relation $P(\hat{a}_i = a_i) = \frac{1}{2} + \varepsilon$ with $\varepsilon > 0$; second, fast correlation attack phase which aims at recovering the secret initial state of LFSR A. Here we adopt the BSC (binary symmetric channel) model for fast correlation attack, as shown in Figure 1.

Our main idea is to regard the sequence $\hat{a}$ as the noisy version of sequence a through the binary symmetric channel representing the noise introduced by the shrinking generator, i.e. $1 - p = P(\hat{a}_i = a_i)$, given p as the crossover probability in the BSC. W.l.o.g assume $p < 0.5$. Our aim is to restore sequence a from $\hat{a}$ by efficient fast correlation attack techniques. Note that several new efficient fast correlation attacks on stream ciphers are proposed recently, [2, 3, 15, 16], enabling us to construct an efficient fast correlation attack on the shrinking generator, which is impossible by traditional techniques. In this paper, we follow

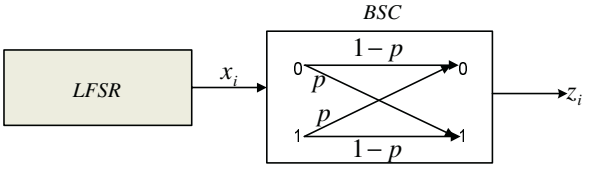

Fig. 1. Model for fast correlation attack.

the method in [3] to mount our attack on the shrinking generator. In nature, our correlation analysis has nothing to do with the decoding algorithm which means other decoding techniques may also be applied, as discussed in Section 4.

The original idea of correlation analysis phase goes back to [21]. We made crucial improvements to the initial method. For simplicity, assume that both the LFSR sequences generated by LFSR A and LFSR S are purely random (a sequence of independent uniformly distributed random variables is called purely random). Consider the probability that z_k equals a_r in the (SG). It is obvious that $k \leq r$. If we regard the event that $s_i = 1$ as success, then the event that z_k equals a_r is equivalent to the event that the kth success of sequence s occurs at the rth trial which obeys the Pascal Distribution. Thus the probability that z_k equals a_r is:

$$P(z_k = a_r) = \binom{r}{k}(\frac{1}{2})^{r+1}. \tag{1}$$

On the other hand, if a_r appears in the keystream z, the following equation holds:

$$a_r = z_{\sum_{i=0}^{r-1} s_i}. \tag{2}$$

When r grows large, the distribution of the sum $\sum_{i=0}^{r-1} s_i$ approximates the Normal Distribution, i.e.

$$\frac{\sum_{i=0}^{r-1} s_i - r/2}{\sqrt{r/4}} \mapsto N(0,1). \tag{3}$$

Let $I_{r/2} = [r/2 - \alpha\sqrt{r/4}, r/2 + \alpha\sqrt{r/4}]$, here comes our main observation: for arbitrary probability p, there exists a α such that whenever a_r appears in keystream z, the following equation holds:

$$P(\sum_{i=0}^{r-1} s_i \in I_{r/2}) = p. \tag{4}$$

As in [5], we formally define two kinds of intuitive notion of imbalance.

Definition 1. *W.l.o.g, we assume the interval $I_{r/2}$ includes odd number of integers. Let $S_0 = \{z_i | i \in I_{r/2}, z_i = 0\}$, $S_1 = \{z_i | i \in I_{r/2}, z_i = 1\}$, the first kind of imbalance of the interval $I_{r/2}$, $Imb_1(I_{r/2})$, is defined as $|S_1| - |S_0|$, where $|\cdot|$ is the cardinality of a set. If $Imb_1(I_{r/2}) \neq 0$, this interval is said to be imbalanced. See Figure 2.*

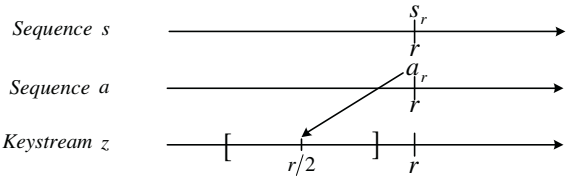

Fig. 2. The interval that a_r probably lies in.

Definition 2. *The notations are the same as those in Definition 1. Let* $P_0^{(r)} = \sum_{z_i \in S_0} P(a_r = z_i)$, $P_1^{(r)} = \sum_{z_i \in S_1} P(a_r = z_i)$, *the second kind of imbalance of the interval* $I_{r/2}$, $Imb_2(I_{r/2})$, *is defined as* $P_1^{(r)} - P_0^{(r)}$. *If* $Imb_2(I_{r/2}) \neq 0$, *this interval is also said to be imbalanced. See Figure 2.*

Now there are two kinds of construction methods of sequence $\hat{a}$ corresponding to these two kinds of imbalance. The first one is a straightforward majority poll according to Definition 1. The second one is a similar but more reasonable poll according to Definition 2.

Method 1. Following Definition 1, if $Imb_1(I_{r/2}) > 0$, let $\hat{a}_r = 1$. Otherwise, let $\hat{a}_r = 0$.

Method 2. Following Definition 2, if $Imb_2(I_{r/2}) \geq 0$, let $\hat{a}_r = 1$. Otherwise, let $\hat{a}_r = 0$.

Both theoretical analysis and experimental results show that sequence $\hat{a}$ constructed above satisfying $P(\hat{a}_i = a_i) = \frac{1}{2} + \varepsilon$ with $\varepsilon > 0$ as expected. We will give the theoretical analysis in next section and the experimental results in Section 4.

Next, we will present a brief description of the fast correlation attack [3] involved in our attack. This attack is a one-pass correlation attack consisting of two stages: pre-processing stage aiming at the construction of parity-check equations of weight k and processing stage in which a majority poll is conducted for D ($D > L - B$) considered bits other than the first B bits $(x_0, x_1, \cdots, x_{B-1})$ of the initial state $(x_0, x_1, \cdots, x_{L-1})$. In general, there are three new ideas proposed in [3]. First, a match-and-sort algorithm is proposed to construct parity-check equations of the following form with respect to a given considered bit x_i

$$x_i = x_{m_1} \oplus \ldots \oplus x_{m_{k-1}} \oplus \sum_{j=0}^{B-1} c_j x_j \tag{5}$$

where m_j $(1 \leq j \leq k-1)$ denote the indices of the keystream bits and the last sum represents a partial exhaustive search over $(x_0, \cdots, x_{B-1})$ of the initial state $(x_0, \cdots, x_{L-1})$. (5) offers plenty of suitable parity-check equations needed for high performance decoding, meanwhile avoids the low weight restriction of the feedback polynomial of the LFSR. Second, after regrouping the parity-check

equations that contain the same pattern of $B - B_1$ initial bits, an application of Walsh transform is suggested to evaluate the parity-check equations in processing stage for a given z_i, i.e. when $\omega = [x_{B_1}, x_{B_1+1}, \cdots, x_{B-1}]$, $F_i(\omega) = \sum(-1)^{t_i^1 \oplus t_i^2}$ is just the difference between the number of predicted 0 and the number of predicted 1, where $t_i^1 = z_{m_1} \oplus \cdots \oplus z_{m_{k-1}} \oplus \sum_{j=0}^{B_1-1} c_j x_j$ and $t_i^2 = \sum_{j=B_1}^{B-1} c_j x_j$. Then for each of the D considered bits, if $F_i(\omega) > \theta$, let $x_i = 0$. If $F_i(\omega) < -\theta$, let $x_i = 1$, where θ is the decision threshold. Third, in order to have at least $L - B$ correctly recovered bits among the D considered bits, a check procedure is used which requires an exhaustive search on all subsets of size $L - B$ among the $L - B + \delta$ bits. The total complexity of the processing stage is:

$$O(2^B D \log_2 \Omega + (1 + p_{err}(2^B - 1)) \binom{L - B + \delta}{\delta} \frac{1}{\varepsilon^2}) \tag{6}$$

where p_{err} is the probability that a wrong guess results in at least $L - B + \delta$ predicted bits and Ω is the expected number of parity-check equations of weight k for each considered bit. For the details of these formulae and the notations, please see the Appendix A and [3].

A summary of our attack is as follows:

1. Input: the feedback polynomial, $f(x)$, of LFSR A, a segment of keystream $z_0, z_1, \cdots, z_{N-1}$, $N' < N$, N' is determined by $N' \approx N - \alpha\sqrt{N'}/2$.
2. Construct sequence $\hat{a} = \hat{a}_0, \cdots, \hat{a}_{N'-1}$ according to Method 1 or Method 2 from keystream $z_0, z_1, \cdots, z_{N-1}$.
3. For each guess of $(a_0, \cdots, a_{B-1})$ and each bit position i, $(i = B + 1, B + 2, \ldots, D)$, evaluate the parity-check equations using the Walsh transform technique. Select those bits passing the majority poll to recover the initial state of LFSR A using the above check procedure.

After having recovered the initial state of LFSR A, we should also restore the initial state of LFSR S. With the knowledge of known sequence of LFSR A and keystream z, the remaining problem is much simplified compared to the original one . One way to do so is to use the method proposed in [6]. Here we do not focus on this problem.

3 Analysis of Our Attack

In this section, we will analyze our attack deeply, mainly on the two correlation analysis methods. We give two theorems on the coincidence probabilities $P(\hat{a}_r = a_r)$ under the above two methods, respectively. We will show that a special case of our method 2 is equivalent to the method proposed by Golić in [6].

3.1 The Coincidence Probability Under Method 1

Keep the assumption that both sequences generated by LFSR A and LFSR S are purely random. Theorem 1 yields the probability that sequence $\hat{a}$ equals sequence a under method 1.

Theorem 1. *Under method 1, the probability that the constructed sequence $\hat{a}$ equals sequence a is given by*

$$P(\hat{a}_r = a_r) = \frac{1}{2} + \frac{1}{2^{2E}}\binom{2E}{E}\frac{p}{4} = \frac{1}{2} + \varepsilon_r. \tag{7}$$

where $2E+1$ satisfying $E = \lfloor(\alpha\sqrt{r} - 1)/2\rfloor$, is the closest odd integer to $\alpha\sqrt{r}$ and $p = \frac{1}{\sqrt{2\pi}}\int_{-\alpha}^{\alpha} e^{-x^2/2}dx$ is the probability in (4).

Proof. According to method 1, we have

$$\begin{aligned}
P(\hat{a}_r = a_r) &= P(s_r = 1)P(\hat{a}_r = a_r|s_r = 1) + P(s_r = 0)P(\hat{a}_r = a_r|s_r = 0) \\
&= \frac{1}{2}P(\hat{a}_r = a_r|s_r = 1) + \frac{1}{4} \\
&= \frac{1}{2}P(\hat{a}_r = a_r|\sum_{i=0}^{r-1} s_i \in I_{r/2}, s_r = 1)P(\sum_{i=0}^{r-1} s_i \in I_{r/2}|s_r = 1) \\
&\quad + \frac{1}{2}P(\sum_{i=0}^{r-1} s_i \bar{\in} I_{r/2}|s_r = 1)P(\hat{a}_r = a_r|\sum_{i=0}^{r-1} s_i \bar{\in} I_{r/2}, s_r = 1) + \frac{1}{4} \\
&= \frac{1}{4} + \frac{1}{4}(1-p) + \frac{p}{2}P(\hat{a}_r = a_r|\sum_{i=0}^{r-1} s_i \in I_{r/2}, s_r = 1) \\
&= \frac{1}{2} - \frac{p}{4} + \frac{p}{2}P^*
\end{aligned}$$

where $P^* = P(\hat{a}_r = a_r|\sum_{i=0}^{r-1} s_i \in I_{r/2}, s_r = 1)$ can be derived by the following equations.

$$\begin{aligned}
P^* &= P(\hat{a}_r = a_r = 0|\sum_{i=0}^{r-1} s_i \in I_{r/2}, \cdot) + P(\hat{a}_r = a_r = 1|\sum_{i=0}^{r-1} s_i \in I_{r/2}, \cdot) \\
&= P(a_r = 0)P(\hat{a}_r = 0|a_r = 0, \sum_{i=0}^{r-1} s_i \in I_{r/2}, s_r = 1) \\
&\quad + P(a_r = 1)P(\hat{a}_r = 1|a_r = 1, \sum_{i=0}^{r-1} s_i \in I_{r/2}, s_r = 1) \\
&= \frac{1}{2}\sum_{i=E}^{2E}\binom{2E}{i}\frac{1}{2^{2E}} + \frac{1}{2}\sum_{i=E}^{2E}\binom{2E}{i}\frac{1}{2^{2E}}.
\end{aligned} \tag{8}$$

(8) comes from the observation that if $a_r = j$ $(j = 0, 1)$, then there must be at least E elements other than a_r itself in $I_{r/2}$ to be j for $\hat{a}_r = a_r = j$ holds. According to $\sum_{i=E}^{2E}\binom{2E}{i} = \sum_{i=0}^{E}\binom{2E}{i}$, we get

$$P^* = \frac{1}{2} + \frac{1}{2^{2E+1}}\binom{2E}{E}.$$

This completes the proof.

Corollary 1. *The coincidence probability $P(\hat{a}_r = a_r)$ is a function of r satisfying*

$$\frac{1}{2} < P(\hat{a}_r = a_r) \leq \frac{3}{4} \tag{9}$$

where the upper bound is achieved when $r = 0$.

Theorem 1 implies that the smaller r, the larger $P(\hat{a}_r = a_r)$ is. Note that our aim is to have a sequence $\hat{a}$ with a large enough correlation to a, which means that we should make the probability $P(\hat{a}_r = a_r)$ as large as possible. The larger ε_r is, the larger number of bits in sequence $\hat{a}$ satisfy $\hat{a}_r = a_r$. However, the above theorem shows that the probability function has an irregular form such that the classical methods for finding global maximum value of regular functions can not be used to obtain its global maximum. Instead, we try to find out the optimum numerical values of $P(\hat{a}_r = a_r)$ for each r. From Theorem 1, we can see that the bias

$$\varepsilon_r = \frac{1}{2^{2E}} \binom{2E}{E} \frac{p}{4} \tag{10}$$

is dependent on the product of p and $\binom{2E}{E}/2^{2E}$. Therefore, the optimum value of ε_r is

$$\varepsilon_{\max}^{(r)} = max_{0 \leq p \leq 1} \{ \frac{1}{2^{2E}} \binom{2E}{E} \frac{p}{4} \}. \tag{11}$$

Note that $2E + 1$ is a measure of the length of $I_{r/2}$ which is determined by the probability p chosen in advance. In intuitive point of view, we should always choose p (by choosing α) rather large so that we can guarantee the interval $I_{r/2}$ always includes the indices of the elements that lie in keystream z. One easy way to do so is to choose p equals to one fixed value such as $0.90, 0.95, \cdots$, even $p = 0.99$. However, both theoretical and experimental results show that the bias ε_r drops so rapidly in this way that the average coincidence probability found is not good enough for an efficient fast correlation attack. Instead, we programmed in Mathematica to find each α that results in $\varepsilon_{\max}^{(r)}$. Figure 3 (In Figure 3, the horizontal axes represents α) shows for each r, where the optimum of α is located in the range $(0, 5)$.

Note that our construction method of sequence $\hat{a}$ is independent of the concrete LFSR structure under the purely random assumption, which means the pre-computation of the optimum values of α would be applied to arbitrary LFSR. Figure 3 shows that the optimum values of α satisfy $1 \leq \alpha \leq 2$ for $r \geq 244$. Noting the instruction Findminimum in Mathematica can only find the local minimum, we use the following two instructions to find the optimum value of α (a represents α):

Findminimum$[-\frac{\binom{2E}{E}}{2^{2E}} \frac{\int_{-a}^{a} e^{-x^2/2} dx}{4\sqrt{2\pi}}, \{a, 0, 5\}]$, $0 \leq r \leq 243$

or

Findminimum$[-\frac{\binom{2E}{E}}{2^{2E}} \frac{\int_{-a}^{a} e^{-x^2/2} dx}{4\sqrt{2\pi}}, \{a, 1, 5\}]$, $r \geq 244$.

Figure 4 (In Figure 4 and 5, the horizontal axis represent keystream length N) shows the locations of the optimum values of α. With the knowledge of the

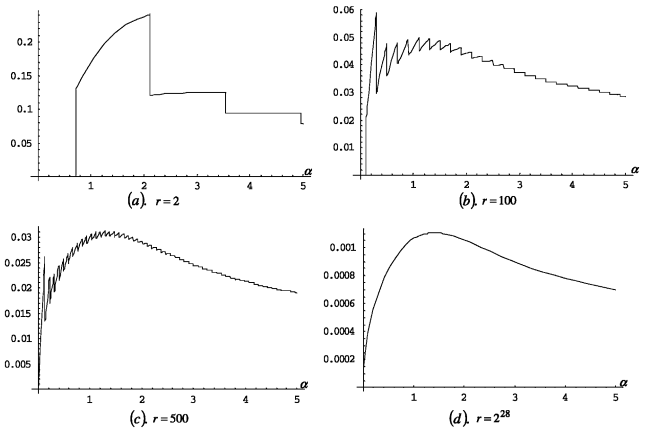

Fig. 3. The optimum position of α.

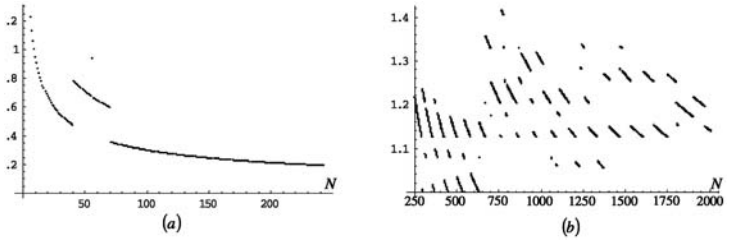

Fig. 4. The optimum value α that results in $\varepsilon_{\max}^{(r)}$. (a)-small scall, (b)-larger scale.

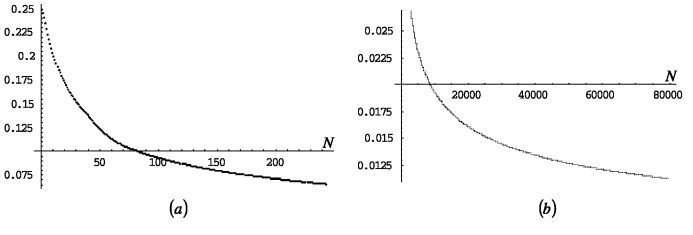

Fig. 5. The values of $\varepsilon_{\max}^{(r)}$. (a)-small scale, (b)-large scale.

optimum values of α, the biases we found are plotted in Figure 5. Let $H = \{\hat{a}_i |\ i \in \{0, 1, \cdots, N-1\}, \hat{a}_i = a_i\}$, the correlation found in this way is defined as $|H|/N$. We can see that the correlations is good enough for an efficient fast correlation attack against LFSR of moderate length. For example, for N=243, it amounts to 0.56555. For $N = 3000$, the correlation is 0.52748 and for $N = 8000$, it is 0.52075. See Section 4.

3.2 The Coincidence Probability Under Method 2

Next, we consider the probability $P(\hat{a}_r = a_r)$ under the construction of method 2. We will show that a special case of method 2 is equivalent to the method proposed by Golić in [6] in a sense that the numerical biases found under both methods (a special case of our method 2 and the method in [6]) are almost the same.

First note that from Definition 2 and (1), we have

$$P_0^{(r)} = \sum_{z_i \in S_0} P(a_r = z_i) = \sum_{z_i \in I_{r/2}} \binom{r}{i} (1 - z_i)(\frac{1}{2})^{r+1} \tag{12}$$

$$P_1^{(r)} = \sum_{z_i \in S_1} P(a_r = z_i) = \sum_{z_i \in I_{r/2}} \binom{r}{i} z_i (\frac{1}{2})^{r+1}. \tag{13}$$

(12) and (13) imply that

$$E(P_0^{(r)}) = E(P_1^{(r)}) = \frac{1}{2} \sum_{z_i \in I_{r/2}} \binom{r}{i} (\frac{1}{2})^{r+1} = \frac{1}{2}(P_1^{(r)} + P_0^{(r)}), \tag{14}$$

where $E(\cdot)$ is the mathematical expected value of the random variable. Note that method 2 actually takes into account the weight (the probability $P(a_r = z_k)$ associated with the point) of each point in $I_{r/2}$ upon making a majority poll, while in method 1, we regard each point in $I_{r/2}$ as the same, i.e. no one is more important than any other one. Therefore,

$$\begin{aligned} P(\hat{a}_r = a_r) &= P(\hat{a}_r = a_r, \sum_{i=0}^{r-1} s_i \in I_{r/2}) + P(\hat{a}_r = a_r, \sum_{i=0}^{r-1} s_i \bar{\in} I_{r/2}) \\ &= \frac{1}{2} + \{\max(P_1^{(r)}, P_0^{(r)}) - \frac{1}{2}(P_1^{(r)} + P_0^{(r)})\} \\ &= \frac{1}{2} + \{\max(P_1^{(r)}, P_0^{(r)}) - E(\max(P_1^{(r)}, P_0^{(r)}))\} = \frac{1}{2} + \varepsilon_r. \end{aligned} \tag{15}$$

Now we consider an important case of method 2. Let $I_{r/2} = \{0, 1, \cdots, r\}$ such that $P_1^{(r)} + P_0^{(r)} = \frac{1}{2}$, i.e. the probability that a_r lies in the interval $I_{r/2}$ is 0.5, instead of 1, due to the nature difference between method 1 and method 2. In this case, $E(P_0^{(r)}) = E(P_1^{(r)}) = \frac{1}{4}$. It follows from (14) and (15) that

$$\begin{aligned} E(\varepsilon_r) &= E(\max(P_1^{(r)}, P_0^{(r)})) - \frac{1}{4} \\ &= E((P_1^{(r)} + P_0^{(r)})/2 + |P_1^{(r)} - P_0^{(r)}|/2) - \frac{1}{4} \\ &= E(|P_1^{(r)} - \frac{1}{4}|). \end{aligned} \tag{16}$$

Since $I_{r/2} = \{0, 1, \cdots, r\}$, we regard $P_1^{(r)} = \sum_{i=0}^{r} \binom{r}{i} z_i (\frac{1}{2})^{r+1}$ as the sum of $r + 1$ independent random variables $\xi_0, \xi_1, \cdots, \xi_r$ satisfying $P(\xi_i = 0) = P(\xi_i =$

$\binom{r}{i}(\frac{1}{2})^{r+1}) = 0.5$. When $r \to \infty$, $P_1^{(r)}$ follows the Normal Distribution, i.e. $P_1^{(r)} \rightarrowtail N(\frac{1}{4}, \sigma^2)$, where the variance $\sigma^2 = \sum_{i=0}^{r} \binom{r}{i}^2 (\frac{1}{2})^{2r+2} \frac{1}{4} = \binom{2r}{r}(\frac{1}{2})^{2r+2}\frac{1}{4}$. Hence, we get

$$E(\varepsilon_r) = \frac{2\sigma}{\sqrt{2\pi}} = \frac{\sqrt{(\frac{1}{2})^{2r}\binom{2r}{r}}}{2\sqrt{2\pi}} \approx \frac{1}{2\sqrt{2\pi}\sqrt[4]{\pi}} \cdot \frac{1}{\sqrt[4]{r}} \approx 0.149828\frac{1}{\sqrt[4]{r}}. \quad (17)$$

Note that the corresponding bias found in [6] is $0.1515\frac{1}{\sqrt[4]{r}}$ based on approximating a binomial distribution by a uniform distribution. Both estimations are almost the same. From above, we get the following theorem.

Theorem 2. *Under method 2 and let $I_{r/2} = \{0, 1, \cdots, r\}$, the probability that the constructed sequence $\hat{a}$ equals sequence a is given approximately by*

$$P(\hat{a}_r = a_r) \approx \frac{1}{2} + 0.149828\frac{1}{\sqrt[4]{r}} \quad (18)$$

where $I_{r/2}$ is the same notation as that defined in Section 2.

Note that we obtain Theorem 2 under a special case of method 2. As in Theorem 1, we also want to maximize the probability $P(\hat{a}_r = a_r)$ under the general case of method 2. In nature, the maximization problem is to determine how long the interval $I_{r/2}$ should be chosen (by choosing α) such that the second kind of imbalance, $Imb_2(I_{r/2})$, can be maximized. The detailed analysis appears to be difficult, for the Normal Distribution may not be used in this case. We just leave this problem open. In the following, we will show that the coincidence probability obtained under Theorem 1 is approximately comparable to those got in Theorem 2 and in [6]. See Table 1. Note that the biases listed in Table 1 are not the average values, which are listed in Section 4. We can see that the bias values got from two methods are very close. Actually, such close values have almost the same inflect on the complexity of the whole fast correlation attack. Hence, any one of them can be used in practice. If all the binomial coefficients $\binom{i}{k}$ $0 \le i \le r$ are pre-computed as suggested in [6] using the recursion $\binom{i}{k} = \binom{i-1}{k-1} + \binom{i-1}{k}$ in $O(i^2)$ time and stored in $O(r^2)$ space, then method 2 will give a slightly higher coincidence. If the optimum values of α have been pre-computed in advance, method 1 is OK.

In addition, from Theorem 2 we can see that with the increase of r, the coincidence probability $P(\hat{a}_r = a_r)$ tends to 0.5 slowly. This fact can be interpreted as the reasonable result of basic design criterion of stream ciphers that the keystream z should satisfy $P(z = 0) = P(z = 1) = 0.5$ and the fact that a binomial distribution approximates a uniform distribution when $r \to \infty$.

Table 1. The one-point bias values of two methods.

r	1000	4000	8000	20000
Th. 1	0.0258843	0.018021	0.0150915	0.0119576
Th. 2	0.0266436	0.0188399	0.0158424	0.012599

4 Experimental Results

In this section we present some simulation results of our attack together with some comparisons with other known attacks on the shrinking generator. The experiments were done on a Pentium 4 PC processor.

First, we list the optimum values of α that give $\varepsilon_{\max}^{(r)}$ in Table 2. We use Mathematica to pre-compute these values in about four hours. It can be easily seen that most of the optimum values of α lie in the interval $(1.3, 1.5)$. The average value $\bar{\alpha} = 1.376395$ corresponds to the average probability $\bar{p} = 83.13\%$. It is worth noting that the optimum values of α are applicable to arbitrary LFSRs due to our purely random assumption in Section 2. Table 3 shows the average biases obtained by two theoretical methods and computer simulations. It is obvious that Theorem 1 is preferable when r is small, while Theorem 2 coincides with simulations better and offers a little better correlation when r grows large. The actual values of ε in Table 3 are found based on a shrinking generator with the following two primitive polynomials as the feedback polynomials of LFSR A and LFSR S, respectively: $f_A(x) = 1+x+x^3+x^5+x^9+x^{11}+x^{12}+x^{17}+x^{19}+x^{21}+x^{25}+x^{27}+x^{29}+x^{32}+x^{33}+x^{38}+x^{40}$ $[3, 15, 16, 10]$ and $f_S(x) = 1+x+x^2+x^3+x^4+x^5+x^{42}$ by method 1. The experimental results are in accordance with the theoretical expectations very well.

In order to compare our attack with other known ones, we consider another example of the shrinking generator with the generating LFSR A of length 61, as suggested in [9]. For practical considerations, we assume the length of LFSR S $\approx$ 61. Following the fast correlation attack in Section 2 and Appendix A,

Table 2. The optimum values of α (N=120000).

Domain	Number of α	Percent
$1.0 \sim 1.1$	248	0.2%
$1.1 \sim 1.2$	3139	2.5%
$1.2 \sim 1.3$	4308	3.6%
$1.3 \sim 1.4$	63480	53.0%
$1.4 \sim 1.5$	48221	40.2%
$1.5 \sim 1.6$	365	0.3%
others	239	0.2%
$\bar{\alpha}$ average	1.376395	100%

Table 3. The average biases ε of two methods and simulations.

N	ε(**Th. 1**)	ε(**Th. 2**)	ε(found)
240	0.0667726	0.0512096	0.054167
3000	0.02748	0.0270324	0.02100
8000	0.02075	0.0211382	0.02037
40000	0.0135484	0.014129	0.015650
80000	0.0113329	0.01188	0.012275
140000	0.00982376	0.0103285	0.008700

we choose the attack parameters as follows: $D = 36, \delta = 3, B = 46, k = 5$ for $L = 61$, the keystream length is $N = 140000 \approx 2^{17.1}$ and the coincidence probability is 0.50982376. We use the parity-check equations of weight 5, which can be obtained in $O(2^{43})$ pre-processing time and can be reused in later as many times as desirable. The expected number of parity-check equations for a given bit is $\Omega = 4.88464 \times 10^{14}$ and the probability that one parity-check equation gives the correct prediction is $q = \frac{1}{2}(1 + 0.01964752^4)$. From Appendix A, in order to have $P_1 \geq (L - B + \delta)/D = 0.5$, we choose $t = 2.4423196361 \times 10^{14}$ such that $P_1 \approx 0.500156$ and $P_v \approx 0.999999$. This gives the success probability

$$P_{succ} = \sum_{j=0}^{3} \binom{18}{3} P_v^{18-j}(1 - P_v)^j \approx 99.9\%.$$

The probability of false alarm is negligible in this case. In fact, the probability P_{err} is limited to $P_{err} \approx 7.6 \times 10^{-45}$. Hence, the total processing complexity is

$$2^{46} \cdot 36 \cdot \log_2 \Omega + (1 + p_{err}(2^{46} - 1))\binom{18}{3}\frac{1}{\varepsilon^2} \approx 2^{56.7786}.$$

Table 4 shows the comparisons of different known attacks on the above example shrinking generator.

Table 4. Comparisons of different attacks on the example shrinking generator.

	[13]	[8]	**A.**[12]	**B.**[12]	**C.**[12]	Our attack
Length of z	few	$2^{10.23}$	few	2^{30}	$2^{30} - 2^{40}$	$2^{17.1}$
Complexity	2^{80}	2^{77}	2^{71}	2^{56}	$2^{50} - 2^{40}$	2^{56}
p_{succ}	100%	100%	66%	66%	66%	99.9%

For the detailed discussion of the concrete values in Table 4, see Appendix B. From Table 4, we can see that the attacks in [13], [8] and the attack A in [12] are all with the complexity higher than an exhaustive search. The attacks B and C in [12] are faster than an exhaustive search. But if a very high probability of success is required, we have to repeat the whole attack at least 4 times, which, for the best complexity result in [12], results in a 2^{42} keystream length and 2^{42} complexity. The required keystream length is too long for a 61-stage LFSR. In contract, the keystream length required in our attack is rather small, $2^{17.1}$, and the complexity is comparable to those in [12]. Hence, our attack offers a better trade-off between these parameters. In addition, our attack is better than the recent proposed attack on irregularly clocked generators in [17]. In that paper, a malformed shrinking generator with a LFSR S of length 26 and LFSR A of length 60 is cracked using an exhaustive search over the initial states of LFSR S with $1000000 \approx 2^{20}$ keystream bits. Besides, several fast correlation attack ideas on the (SG) have been proposed in [6]. However, few concrete results are available in that paper, making it difficult to make a comparison with it.

Some Remarks. An important fact about our attack is that the coincidence probability between a and $\hat{a}$ decreases, though rather slowly, with the increasing length of keystream. Hence, we propose two recommendations on attacking the shrinking generator.

1. It is of great importance to improve the fast correlation attack techniques by reducing the number of keystream bits required and deriving more efficient algorithm to construct parity-check equations with a little more weight. A new fast correlation attack is proposed in [18] without the detailed processing procedures, whose main advantage is the small amount of keystream necessary for a success attack with respect to a certain noise level compared to other attacks. From our experiments, the bias corresponding to $N = 3000$ keystream is 0.0274845, we think it is a promising way to apply this kind of attack to the shrinking generator.
2. Another direction is to consider the sequence $\hat{a}$ satisfying $P(\hat{a}_i = a_i) = p_i$ with different p_i, which is more closer to the truth of the construction method. Actually, such a method is used in [14] whose main disadvantage is the weight restriction of the feedback polynomials. Therefore, it is important to develop new fast correlation attacks applicable to the different p_i case, while maintaining the property that it is independent of the feedback polynomial's weight.

5 Conclusions

In this paper, we demonstrate a fast correlation attack on the shrinking generator with fixed connections. Our attack confirms that Golić's conjecture is correct. In addition, comparisons with other known attacks reveal that our attack offers a better trade-off between the required keystream length, success probability and the complexity.

Acknowledgements

We would like to thank the anonymous reviewers for very helpful comments.

References

1. A. Biryukov, "Block Ciphers and Stream Ciphers: The State of the Art", http://eprint.iacr.org/2004/094.pdf.
2. A. Canteaut, M. Trabbia, "Improved Fast Correlation Attacks Using Parity-Check Equations of Weight 4 and 5", *Advances in Cryptology-EUROCRYPT'2000*, LNCS vol. 1807, Springer-Verlag, (2000), pp. 573-588.
3. P. Chose, A. Joux, M. Mitton, "Fast Correlation Attacks: An Algorithmic Point of View", *Advances in Cryptology-EUROCRYPT'2002*, LNCS vol. 2332, Springer-Verlag, (2002), pp. 209-221.
4. D. Coppersmith, H. Krawczyk, Y. Mansour, "The Shrinking Generator", *Advances in Cryptology-Crypto'93*, LNCS vol. 773, Springer-Verlag, (1994), pp.22-39.
5. P. Ekdahl, T.Johansson, "Predicting the Shrinking Generator with Fixed Connections", *Advances in Cryptology-EUROCRYPT'2003*, LNCS vol. 2656, Springer-Verlag, (2003), pp. 330-344.

6. J. Dj. Golić, "Correlation analysis of the shrinking Generator", *Advances in Cryptology-Crypto'2001*, LNCS vol. 2139 Springer-Verlag, (2001), pp. 440-457.
7. J. Dj. Golić, "Computation of Low-weight parity-check ploynomials", *Electronic Letters*, Vol. 32, No. 21, pp. 1981-1982, October 1996.
8. J. Dj. Golić, "Embedding and probabilistic correlation attacks on clock-controlled shift registers", *Advances in Cryptology-EUROCRYPT'94*, LNCS vol. 950, Springer-Verlag, (1994), pp. 230-243.
9. H. Krawczyk, "The shrinking generator: Some practical considerations", *Fast Software Encryption-FSE'94*, LNCS vol. 809, Springer-Verlag, (1994), pp. 45-46.
10. T. Johansson, F. Jonnson, "Improved fast correlation attack on stream ciphers via convolutional codes", *Advances in Cryptology-EUROCRYPT'1999*, LNCS vol. 1592, Springer-Verlag, (1999), pp. 347-362.
11. T. Johansson, F. Jönsson, "Fast correlation attacks through reconstruction of linear polynomials", *Advances in Cryptology-Crypto'2000*, LNCS vol. 1880, Springer-Verlag, (2000), pp. 300-315.
12. T. Johansson, "Reduced complexity correlation attacks on two clock-controlled generators", *Advances in Cryptology-ASIACRYPT'98*, LNCS vol. 1514, Springer-Verlag, (1998), pp. 342-357.
13. A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press,1997.
14. W. Meier, O. Staffelbach, "Fast correlation attacks on certain stream ciphers", *Journal of Cryptology*, (1989) 1 pp. 159-176.
15. M. Mihaljević, P.C. Fossorier, H.Imai, "Fast correlation attack algorithm with list decoding and an application", *Fast Software Encryption-FSE'2001*, LNCS vol. 2355, Springer-Verlag, (2002), pp. 196-210.
16. M. Mihaljević, P.C. Fossorier, H.Imai, "A Low-complexity and high-performance algorithm for fast correlation attack", *Fast Software Encryption-FSE'2000*, LNCS vol. 1978, Springer-Verlag, (2001), pp. 196-212.
17. H. Molland, "Improved Linear Consistency Attack on Irregular Clocked Keystream Generators", *Fast Software Encryption-FSE'2004*, LNCS vol. 3017, Springer-Verlag, (2004), pp. 109-126.
18. M. Noorkami, F. Fekri, "A Fast Correlation Attack via Unequal Error Correcting LDPC Codes", *CT-RSA'2004*, LNCS vol. 2964, Springer-Verlag, (2004), pp. 54-66.
19. L. Simpson, J. Dj. Golić, "A probabilistic correlation attack on the shrinking generator", *ACISP'98*, LNCS vol. 1438, Springer-Verlag, (1998), pp. 147-158.
20. D. Wagner, "A Generalized Birthday Problem", *Advances in Cryptology-Crypto'2002*, LNCS vol. 2442, Springer-Verlag, (2002), pp. 288-303.
21. D. F. Zhang, W. D. Chen, "Information Leak analysing on the Shrinking Generator and the Self-Shrinking Generator", *Journal of China Institute of Communications*, Vol. 17, No. 4, pp. 15-20, July 1996.

A Notations and Formulae of a One-Pass Fast Correlation Attack

1. $P(z_i = x_i) = \frac{1}{2}(1 + \varepsilon)$.
2. N is the length of the keystream.
3. L is the length of the LFSR.
4. B is the number of bits partially exhausitive searched.

5. D is the number of bits under consideration.
6. k is the weight of the parity-check equations.
7. $q = \frac{1}{2}(1 + \varepsilon^{k-1})$ is the probability that one parity-check equation yielding the correct prediction.
8. Ω is the expected number of weight k parity-check equations for each considered bit.
9. δ is the number of bits that predicted other than the $n - B$ bits.
10. $P_1 = \sum_{j=\Omega-t}^{\Omega}(1-q)^{\Omega-j}q^j\binom{\Omega}{j}$ is the probability that at least $\Omega - t$ parity-check equations give the correct result, where t is the smallest integer satisfying $D \cdot P_1 \geq L - B + \delta$.
11. θ is the threshold such that $\theta = \Omega - 2t$.
12. $P_2 = \sum_{j=\Omega-t}^{\Omega}(1-q)^{j}q^{\Omega-j}\binom{\Omega}{j}$ is the probability that at least $\Omega - t$ parity-check equations give the wrong result.
13. $P_v = P_1/(P_1 + P_2)$ is the probability that a bit is correctly predicted with at least $\Omega - t$ parity-check equations give the same prediction.
14. $P_{succ} = \sum_{j=0}^{\delta}\binom{L-B+\delta}{j}P_v^{L-B+\delta-j}(1-P_v)^j$ is the probability that at most δ bits are wrong among the $n - B + \delta$ predicted bits.
15. $E = \frac{1}{2^{\Omega-1}}\sum_{j=\Omega-t}^{\Omega}\binom{\Omega}{j}$ is the probability that a wrong guess yields at least $\Omega - t$ identical predictions for a given bit.
16. $P_{err} = \sum_{j=L-B+\delta}^{D}\binom{D}{j}E^j(1-E)^{D-j}$ is the probability that false alarm occurs.
17. When $k = 4$, the time complexity of the pre-processing stage is $O(N^2 \log N)$. When $k = 5$, the time complexity is $O(DN^2 \log N)$. In both cases, the memory complexities are $O(N)$.

B Remarks on the Concrete Values in Table 4

The attack in [13] is a divide-and-conquer attack on LFSR S requiring $O(2^{L_S}L_A^3)$ operations. For $L_S \approx L_A = 61$, it amounts to 2^{80}. The probabilistic attack proposed in [8] is also an exhaustive attack with complexity around $2^{L_A}(4L_A)^2$. As in [12], here we choose $4L_A$ for unique decoding. For $L_A = 61$, the complexity is 2^{77}. There are three attacks proposed in [12]. Attack A is an exhaustive search using the decoding algorithm given in that paper. Both attack B and C are based on searching for specific weak subsequences in the keystream z. The difference between B and C is that several weak subsequences are required in attack C, which results in the very long length of the required keystream, i.e. 2^{40}. Though the complexity of C is the lowest, 2^{40}, the required keystream length, 2^{40}, is absolutely unrealistic for a LFSR A of length 61. Besides, the decoding algorithm in [12] has a failure probability 0.34, when its complexity is assumed to be 2^{10}.